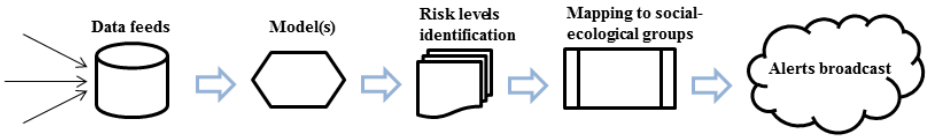
Informatics System Design Considerations

George Mathew, MIT Lincoln Laboratory, Lexington, MA 02420

The alert system to be developed has the structural flow shown in the following figure.



In order to design and develop this system in particular and any informatics system in general, the overall informatics system architecture has to be established.

I. Architectural Considerations

Establishing architectural framework will help lay the foundation for implementation and identify resources required in carrying out the project. The data product design is influenced by scientific factors, data architecture, information architecture, and infrastructure architecture

Scientific factors include finding models that are suitable for the problem, matching the available data feeds to the model builders, and refactoring models for improvements. The exploration and evaluation of data for identifying the right attributes has to be done with the help of domain experts. The model that gives best resolution results has to be identified by practitioners. Operationalizing the models requires programming effort to implement the code in the integrated environment.

Data Architecture is comprised of data collection, fusion, cleansing, and standardization. The attributes in the data feeds and their values have to be identified in a uniform way to be used by analytics. It may be necessary to normalize the values. Information architecture outlines data management foundations at the system level and focuses on the lifecycle management and workflows. It involves plotting data pipelines, identifying data repositories and establishing the data retention/incineration processes. Data retention procedures include regular backups as well as data archival. For an alert system, it is necessary to establish the levels of severity for alerts and the socio-ecological target groups to be associated with each severity level. The score generated by a regression model needs to be mapped to Asthma exacerbation risk levels.

Infrastructure architecture lays the blueprint for the deployment of hardware, software and network components. The technology choices of commodity hardware, databases and software stacks are usually driven by in-house familiarity and comfort level. However, there may be specialized technology selections to be done depending on specificity. For example, a No-SQL database may be a better choice for an unstructured data set compared to a relational database. The software stack includes the backend and frontend components. On the frontend, users could receive alerts via email or text message. Another option is a mobile app for the end user. Backend interfaces are required for application administrators to manage the system. This includes user account management, risk levels management and user roles management. Reporting mechanisms or dashboard for executives may be necessary to be built depending on the requirements of the institution. Hosting the backend systems can be done in-house or in the cloud or as a hybrid. The system components should be monitored for outages.

II. Implementation Stages and Strategies

A typical system implementation has various stages. In general, these stages are:

1. Requirements Gathering
2. Resources Allocation
3. Architecting
4. Development/Integration
5. Pilot testing
6. Production roll out

The requirements for frontend and backend of the system need to be captured. In the architecting stage, these requirements are taken into consideration. Pilot testing is usually done with a selected sample of individuals from each category of system users.

There are two ways to implement the informatics system. One is incremental approach and the other is full scale implementation. In incremental approach, tasks that are carried out manually are automated one by one with a clear understanding of integrating individual components as the project progresses. In full scale implementation, a holistic approach to automation is adopted.